ISIS - Recommendation #4495

Does pixel2map require a UniversalGroundMap?

2016-10-28 10:50 AM - Jesse Mapel

Status:	Closed		
Priority:	Normal		
Assignee:	Ian Humphrey		
Category:	Applications		
Target version:	FY17 Sprint 3		
Impact:	No impact to user; removed an unnecessary UniversalGroundMap instances from pixel2map.	**Software Version:**	

Description

pixel2map creates a UniversalGroundMap for each input cube before starting processing. The ground map is not passed in anywhere and produces the following warning message when compiling pixel2map:

```
warning: variable 'groundMap' set but not used [-Wunused-but-set-variable]
   UniversalGroundMap *groundMap;
                       ^
```

Does pixel2map need this ground map? If not remove it.

History

#1 - 2016-10-28 12:23 PM - Jesse Mapel

I tested this on some Dawn VIR test data and the instantaneous FOV case was not affected by removing it

#2 - 2016-10-28 03:40 PM - Tammy Becker

- Status changed from New to Acknowledged

#3 - 2016-11-04 12:59 PM - Jesse Mapel

groundmap was changed back to g_groundmap and made a global variable to prevent build warnings.

#4 - 2016-11-10 10:17 AM - Jason Laura

- Target version set to FY17 Backlog

#5 - 2016-11-10 11:52 AM - Jason Laura

- Story points set to 1

#6 - 2016-11-10 11:52 AM - Jason Laura

- Target version changed from FY17 Backlog to FY17 Sprint 3

#7 - 2016-11-17 04:01 PM - Ian Humphrey

- Assignee set to Ian Humphrey

Assigned to myself as I have been digging around in pixel2map (#4433, #4520).

This can be closed very quickly, I just need time to add a write-up to this ticket.

I will most likely integrate changes for this ticket into the #4433 checkout.

#8 - 2016-11-18 11:29 AM - Ian Humphrey

- Status changed from Acknowledged to In Progress

#9 - 2016-11-18 12:40 PM - Ian Humphrey

- Status changed from In Progress to Resolved

- Impact updated

I will remove the references to pixel2map's UniversalGroundMap in my changes for ticket [#4433](#4433).

pixel2map does not require a UniversalGroundMap - it is an unused variable.

---

pixel2map does not need to set up a UniversalGroundMap as one is set up and used by the ProcessGroundPolygons instance in pixel2map.cpp.

In pixel2map, there is a call on the ProcessGroundPolygons that sets up the two output cubes (the TO cube and the -count- cube).

```
g_processGroundPolygons.SetStatCubes("TO", pvl, bands);
```


Internally, this SetStatCubes() method calls another ProcessGroundPolygons member:

```
ProcessGroundPolygons::SetStatCubes(const QString &parameter, Pvl &map, inst bands) {
  ...
  SetStatCubes(avgString, countString, atts, map, bands);
}
```


Which calls another overloaded version of SetStatCubes() which sets up an internal UniversalGroundMap member:

```
ProcessGroundPolygons::SetStatCubes(const QString &avgFileName,
                                    const QString &countFileName,
                                    CubeAttributeOutput &atts,
                                    Pvl &map,
                                    int bands) {
  ...
  /*We need a ground map for converting lat/lon to line/sample  see Convert()*/
  p_groundMap = new UniversalGroundMap(*OutputCubes[0]);
}
```


This UniveralGroundMap member is used for rasterizing the input cube pixels:

pixel2map.cpp

```
// for each input image
   ...
   processBrick.StartProcess(rasterizePixel);

void rasterizePixel(Isis::Buffer &in) {
  vector<double>lat, lon;
  vector<double>dns;
  for each value in buffer:
    dns.push_back(value);
  // Get IFOVs in lat/lon space
  // loop through each ifov list
  ...
    //rasterize ifov if > 3 vertices (but doesn't a polygon only need 3?)
    g_processGroundPolygons.Rasterize(lat,lon,dns);

...}

ProcessGroundPolygons::Rasterize(lat, lon, dns) {
  Convert(lat, lon); // USES the set-up UniversalGroundMap

...}
```

Therefore, we do not need this member as ProcessGroundPolygons::SetStatCubes() call in pixel2map sets this up for us.

#10 - 2016-11-18 02:47 PM - Makayla Shepherd

- Status changed from Resolved to Closed